Improving Processes across Three CMMs

Case Study: Requirements Processes

Linda Ibrahim
November 1996

This document describes an approach to improving combined processes that cross over more than one Capability Maturity Model (CMM). That approach is partially illustrated with "Requirements" processes that occur across different AMS life cycle phases..

Contents

Appendix

1. Introduction

This document has two purposes:

- to describe an approach to improving combined processes that cross over more than one Capability Maturity Model (CMM) and that occur across different AMS life cycle phases
- to illustrate this approach with what we denote the "Requirements" processes

The CMMs under consideration are those most directly related to FAA's acquisition of software-intensive systems: the Systems Engineering CMM (SE-CMM), the Software Acquisition CMM (SA-CMM) and the CMM for Software (SW-CMM).

The ideas put forth here build on the recommendation to improve related processes together across the CMMs, and to improve them where they occur across the AMS Life Cycle. *(For* *more information on this approach, see "An Analysis of Three Capability Maturity Models and their Relationship to the Acquisition Management System" [Ibrahim, Oct. 96]).*

2. Improvement Goals and Strategy

2.1 General Improvement Goals across CMMs

A project focusing on improving a combined process area would seek several improvement goals.

FAA Improvement

FAA “in house” improvement goals would include:

- FAA Systems Engineering Improvement *(where FAA carries out systems engineering activities):*
  Achieve a certain capability level for the SE-CMM process area

- FAA Software Acquisition Improvement:
  Satisfy the Goals for the SA-CMM key process area

- FAA Software Engineering Improvement *(where FAA carries out software development and/or maintenance responsibilities)*
  Satisfy the Goals for the SW-CMM key process area

Supplier Improvement

The FAA might also seek improvement in the related processes carried out by FAA suppliers. Improving supplier process capability might involve appraising supplier processes as a part of source selection or examining those processes in post-award situations using some appraisal method such as capability evaluations in combination with risk management. If the FAA chose to follow that path, then similar improvement goals might be sought, e.g.:

- FAA *Supplier* Systems Engineering Improvement *(where FAA supplier carries out systems engineering activities):*
  Achieve a certain capability level for the SE-CMM process area

- FAA *Supplier* Software Engineering Improvement: *(where FAA supplier carries out software development and/or maintenance responsibilities)*
  Satisfy the Goals for the SW-CMM key process area

2.2 Systems Engineering Improvement Goals via the SE-CMM

Systems engineering improvement, as measured by the SE-CMM, means ascending the SE-CMM capability levels from Level 0 (not performed) to Level 5 (continuously improving) for the base practices in the improvement area. These levels are shown below.

SE-CMM Capability Level	*Common Features*	*Generic Practices*
Level 0 - Not Performed *General failure to perform the base practices of the process area*	no common features	none
Level 1 - Performed Informally *Base practices of the process area are generally performed*	1.1 Base Practices are performed	1.1.1 Perform the process
Level 2 - Planned and Tracked *Base practices of the process area are planned and tracked*	2.1 Planning Performance 2.2 Disciplined Performance 2.3 Verifying performance 2.4 Tracking performance	2.1.1 Allocate resources 2.1.2 Assign responsibilities 2.1.3 Document the process 2.1.4 Provide tools 2.1.5 Ensure training 2.1.6 Plan the process 2.2.1 Use plans standards & procedures 2.2.2 Do config management 2.3.1 Verify process compliance 2.3.2 Audit work products 2.4.1 Track with measurement 2.4.2 Take corrective action
Level 3 - Well Defined *Base practices are performed according to a well-defined process using approved, tailored versions of standard documented processes.*	3.1 Defining a Standard Process 3.2 Perform the Defined Process	3.1.1 Standardize the process 3.1.2 Tailor the standard process 3.2.1 Use a well-defined process 3.2.2 Perform defect reviews 3.2.3 Use well-defined data
Level 4 - Quantitatively Controlled *Detailed measures of performance are collected and analyzed.*	4.1 Establishing Measurable Quality Goals 4.2 Objectively Managing Performance	4.1.1 Establish quality goals 4.2.1 Determine process capability 4.2.2 Use process capability
Level 5 - Continuously Improving *The organization establishes quantitative performance goals for process effectiveness and efficiency based on its business goals.*	5.1 Improving Organizational Capability 5.2 Improving Process Effectiveness	5.1.1 Establish process effectiveness goals 5.1.2 Continuously improve the standard process 5.2.1 Perform Causal analysis 5.2.2 Eliminate defect causes 5.2.3 Continuously improve the defined process

2.3 Software Acquisition and Software Engineering Improvement via the SA-CMM and the SW-CMM

Improvement according to the SA-CMM and the SW-CMM means satisfying the goals for the key process area (KPA) where improvement is being sought. In order to satisfy the goals, the key practices (or acceptable alternative practices) need to be in place. The key practices, for each key process area, are organized into 5 common features:

SW-CMM and SA-CMM Common Features	*Generic Description*
Commitment to Perform	Actions the organization must take to ensure that the process is established and will endure, typically establishing organizational policies and leadership.
Ability to Perform	Preconditions that must exist in the project or organization to implement the process competently, such as resources, organizational structures, & training.
Activities Performed	Activities, roles, and procedures necessary to implement a KPA, such as establishing plans and procedures, performing the work, tracking it, and taking corrective action as necessary.
Measurement and Analysis	Basic measurement practices to determine status related to the process, used to control and improve the process.
Verifying Implementation	Steps to ensure the activities are performed in compliance with the process, typically by means of reviews, audits, or quality assurance.

Organizational capability improvement, according to these CMMs, means satisfying KPAs in the specific leveled order prescribed by the models e.g. satisfy level 2 KPAs and you've achieved level 2, satisfy level 2 and level 3 KPAs and you've achieved level 3, etc.

2.4 Combining the Improvement Approaches

Levels 0 and 1: Level 0 and Level 1 of the SE-CMM have no direct counterpart in the SA-CMM / SW-CMM improvement architecture. Thus we propose adopting the SE-CMM levels. Thus obtaining Level 1 means that the relevant base practices and "activities performed" practices are performed.

Level 2: Carrying out the Commitment to Perform, Ability to Perform, Measurement and Analysis, and Verifying Implementation practices in the SW-CMM and SA-CMM models is similar to carrying out the Level 2 Common features of the SE-CMM. Thus for simplicity we propose adoption of the SE-CMM level 2 common features as inclusive of both approaches, with the addition of an "ensuring commitment" common feature with generic practices as follows:

2.0 Ensuring commitment	2.0.1 Document the policy
	2.0.2 Designate responsibility

Then, the improvement task directed at satisfying KPA goals and achieving level 2 can be seen as applying those generic practices to the base practices of the relevant SE-CMM process area and the "activities performed" key practices of the relevant SW-CMM and SA-CMM KPAs.

Levels 3, 4 and 5: The CMM architectures directly differ on what these capability levels denote. Since the scope of this discussion is improvement of a combined process area (rather than organizational improvement) the SE-CMM capability levels would guide improvement of a specific process beyond level 2.

2.5 General improvement strategy

An improvement team chartered to improve a cross-CMM process area across the AMS Life Cycle would typically proceed as follows:

1. Become familiar with the relevant process areas as described in the CMMs

- analyze the base practices, the goals, and the activities performed key practices
- identify overlaps and relationships between/among the practices
- derive a combined list of practices

2. Become familiar with the current state of the relevant process in relation to AMS

- look at AMS life cycle phases and activity areas where these practices might be performed
- analyze AMS process information (guidelines, policies, charts, templates, ... as available)
- find out who carries out the processes across the life cycle

3. Document the current process

- describe current process
- measure current process

4. Analyze current process

- relate current process to combined list of best practices from the CMMs
- identify additional practices or missing practices; identify practices essential for FAA success (regardless of practices contained in the CMMs)
- relate that desired process to generic practices of desired capability level
- identify areas for improvement; prioritize
- provide recommendations to chartering group

5. Improve the process according to agreed recommendations

- plan for improvement (e.g. pilot strategy, pilot training, etc. or full rollout)
- carry out selected improvement recommendations
- measure improved process

Note that institutionalization and monitoring of improved processes are beyond the scope of the improvement team charter as envisioned here. Those responsibilities are left to the body that charters the improvement team (for example the Software Engineering Process Group (SEPG) and/or the Systems Engineering Best Practice Team (SEBPT)).

3. Application to “Requirements” Improvement

This section looks at the first few steps that an improvement team might carry out to improve “Requirements” processes across the AMS.

“Requirements” combines the following process areas:

- “Derive and Allocate Requirements” process area of the SE-CMM,
- “Requirements Development and Management” key process area of the SA-CMM,
- “Requirements Management” key process area of the SW-CMM, and
- “Software Product Engineering” key process area of the SW-CMM, relevant activities only

The assumption is that we will deal only with FAA improvement, not supplier improvement.

3.1 Requirements Improvement Goals

- FAA Systems Engineering Improvement *(where FAA carries out systems engineering activities):*

  Achieve capability level 2 (planned and tracked) *[or possibly capability level 3 (well-defined*)] for the SE-CMM process area “Derive and Allocate Requirements”
- FAA Software Acquisition Improvement:

  Satisfy the Goals for the SA-CMM key process area “Requirements Development and Management” which are:

  1. Software-related contractual requirements are developed and maintained in conjunction with the end user and other affected groups.

  2. Software-related contractual requirements are unambiguous, traceable, and verifiable.

  3. The software-related contractual requirements baseline is established and managed.
- FAA Software Engineering Improvement *(where FAA carries out software development and/or maintenance responsibilities)*

  Satisfy the Goals for the SW-CMM key process area “Requirements Management” which are:

  1. System requirements allocated to software are controlled to establish a baseline for software engineering and management use.

  2. Software plans, products, and activities are kept consistent with the system requirements allocated to software.

  Satisfy the Goals for the SW-CMM key process area “Software Product Engineering” as pertaining to “requirements” which are:

  1. The software engineering tasks are defined, integrated, and consistently performed to produce the software.

  2. Software work products are kept consistent with each other.

Achieving these combined improvement goals, according to the approach advocated in section 2, would mean carrying out the combined practices of the process areas according to the following common features and generic practices:

Common Feature	*Generic Practices*	
2.0 Ensuring commitment 2.1 Planning Performance 2.2 Disciplined Performance 2.3 Verifying performance 2.4 Tracking performance	2.0.1 Document the policy 2.0.2 Designate responsibility 2.1.1 Allocate resources 2.1.2 Assign responsibilities 2.1.3 Document the process 2.1.4 Provide tools 2.1.5 Ensure training	2.2.1 Use plans standards & procedures 2.2.2 Do config management 2.3.1 Verify process compliance 2.3.2 Audit work products 2.4.1 Track with measurement 2.4.2 Take corrective action

	2.1.6 Plan the process

3.2 Combining the “Requirements” practices

Analyzing the 23 base practices and “activities performed” practices from the 3 CMM process areas, and combining the activities when deemed appropriate, results in 10 requirements practices as shown below.

Combined practice name	***SE-CMM*** **Derive and Allocate Requirements:** ***Base Practices***	***SA-CMM*** **Requirements Development and Management:** ***Activities Performed***	***SW-CMM*** **Requirement Management:** ***Activities Performed***	***SW-CMM*** **Software Product Engineering:** ***selected Activities Performed***
(these activities are covered by generic practices 2.2.1 and 2.1.4)		1. The project team performs its activities in accordance with its documented reqmts development and management *plans.*		1. Appropriate software eng. *methods and tools* are integrated into the project’s defined software process.
1. Develop operational concept	2.1 Develop a detailed operational concept of interaction of system, user, and environment, that satisfies the operational need.			
2. Identify key requirements	2.2 Identify key requirements that have a strong influence on cost, sched, functional-, risk,or performance			
3. Partition functions	2.3 Partition functions into groups based on established criteria to facilitate & focus reqmts analysis			
4. Derive requirements	2.4 Derive, from the system and other reqmts, requirements that may be logically inferred and implied as essential to system effectiveness			
5. Identify interface requirements	2.5 Identify the *requirements* assoc. with external *interfaces* to the system and interfaces between functional partitions or objects			

Combined practice name	***SE-CMM*** **Derive and Allocate Requirements:** ***Base Practices***	***SA-CMM*** **Requirements Development and Management:** ***Activities Performed***	***SW-CMM*** **Requirement Management:** ***Activities Performed***	***SW-CMM*** **Software Product Engineering:** ***selected Activities Performed***
6. Allocate and analyze requirements	2.6 Allocate requirements to functional partitions, objects, people, or support elements to support synthesis of solutions 2.7 Analyze requirements to ensure they are verifiable by the methods available to the development effort.		1. The software engineering group reviews the allocated requirements before they are incorporated into the software project.	2. The *software requirements* are developed, maintained, documented, and verified by systematically analyzing the allocated requirements according to the project's defined software process
7. Capture and baseline requirements	2.9 Capture system and other requirements, derived requirements, derivation rationale, allocations, traceability, and requirements status.	2. The project team baselines the software-related contractual requirements and places them under change control early in the project, but not later than release of solicitation package		
8. Use requirements in development / maintenance			2. The software engineering group uses the allocated requirements as the basis for software plans, work products, and activities	3. The software design is developed, maintained, documented, and verified, according to project's defined software process, to accommodate *the software requirements* and to form the framework for coding. 4.The software code is developed, maintained, documented and verified, according to project defined software process,to implement *software requirements* and software design.

Combined practice name	***SE-CMM*** **Derive and Allocate Requirements:** ***Base Practices***	***SA-CMM*** **Requirements Development and Management:** ***Activities Performed***	***SW-CMM*** **Requirement Management:** ***Activities Performed***	***SW-CMM*** **Software Product Engineering:** ***selected Activities Performed***
				7. System and acceptance testing of the software are planned and performed to demonstrate that *the software satisfies its requirements.*
9. Appraise and incorporate requirements changes		3. The project team appraises system requirements change requests for their impact on the software being acquired. 4. The project team appraises all changes to the software-related contractual requirements for their impact on performance, architecture, supportability, and system resource utilization and on contract schedule and cost.	3. Changes to the allocated requirements are reviewed and incorporated into the software product.	
10. Ensure consistency and traceability	2.8 Maintain requirements traceability to ensure that lower level (derived) requirements are necessary and sufficient to meet the objectives of higher level requirements.	5. Bi-directional traceability between the software-related contractual requirements and the contractor's software work products and services is maintained throughout the development.		10. Consistency is maintained across software work products, including the software plans, process descriptions, *allocated requirements*, *software requirements*, software design, code, test plans, and test procedures.

3.3 Mapping the Combined Practices to the AMS

Next, those 10 requirements practices would be mapped to the AMS to determine where they would be performed, and by whom. Additional practices would also be identified. We don't carry out that exercise here, but present a matrix which could capture those results. We also provide, in Appendix A2, some draft AMS process information on Planning and Management, System Procurement, and System Engineering during the solution implementation, in-service management, and service-life extension phases of AMS. That plus other AMS-related information would help complete this mapping.

Requirements Practices/ Activities	*FAA Acquisition Life Cycle Phases and Major Activities*											
	Mission Analysis	*Investment Analysis*		*Solution Implementation*				*In-Service Management*			*Service-Life Extension*	
	Ident. Need	*Req mts*	*Inv mt*	*Plan*	*Obt ain*	*Dep loy*	*Review*	*Sust ain*	*Ass ess*	*Exte nd*	*Inv An.*	*Rem ove*
1. Develop operational concept												
2. Identify key requirements												
3. Partition functions												
4. Derive requirements												
5. Identify interface requirements												
6. Allocate and analyze requirements												
7. Capture and baseline requirements												
8. Use requirements in development / maintenance												
9. Appraise and incorporate requirements changes												
10. Ensure consistency and traceability												
other practices essential for FAA success ...												

4. Conclusions

This discussion has shown that:

- it is possible to devise a combined improvement method which captures the different approaches advanced by the various CMMs
- this method can be applied to improvement of processes across CMMs.
- related practices across the CMM processes can be combined leading to more efficiency in process improvement efforts

It is recommended that this approach be used to effect "Requirements" and other process improvements synergistically across the systems engineering, software acquisition, and software engineering disciplines and across the AMS life cycle processes carried out by FAA.

Appendix

A1. Summary of Process Areas Related to "Requirements"

SE-CMM Process Area 2: Derive and Allocate Requirements

Process Area - purpose	*Base Practices*
2. **Derive and Allocate Requirements -** to analyze the system and other requirements and derive a more detailed and precise set of requirements. These derived requirements are allocated to system functions; objects; people; and supporting processes, products, and services, which can be used to synthesize solutions. This process area addresses both the analysis of system-level requirements and the allocation of system-level or derived requirements to lower level functions or objects. This involves addressing the concept of operations, functional partitioning, object identification, and performance allocation, as well as capturing the status and traceability of requirements. The derived and allocated requirements will evolve as the systems requirements evolve over time. When corrective actions have an impact on requirements, it may be necessary to revise the derived and allocated requirements.	2.1 Develop a detailed operational concept of the interaction of the system, the user, and the environment, that satisfies the operational need. 2.2 Identify key requirement that have a strong influence on cost, schedule, functionality, risk, or performance. 2.3 Partition functions into groups based on established criteria (such as similar functionality, performance, or coupling) to facilitate and focus the requirements analysis. 2.4 Derive, from the system and other (e.g. environmental) requirements, requirements that may be logically inferred and implied as essential to system effectiveness. 2.5 Identify the requirements associated with external interfaces to the system and interfaces between functional partitions or objects. 2.6 Allocate requirements to functional partitions, objects, people, or support elements to support synthesis of solutions. 2.7 Analyze requirements to ensure that they are verifiable by the methods available to the development effort. 2.8 Maintain requirements traceability to ensure that lower level (derived) requirements are necessary and sufficient to meet the objectives of higher level requirements. 2.9 Capture system and other requirements, derived requirements, derivation rationale, allocations, traceability, and requirements status.

SA-CMM Key Process Area: Requirements Development and Management

KPA - purpose	*Goals*	*Activities*
Requirements Development and Management - to establish a common and unambiguous definition of software-related contractual requirements that is understood by the project team, end user, and the contractor.	1. Software-related contractual requirements are developed and maintained in conjunction with the end user and other affected groups. 2. Software-related contractual requirements are unambiguous, traceable, and verifiable. 3. The software-related contractual requirements baseline is established and managed.	1. The project team performs its activities in accordance with its documented requirements development and management plans. 2. The project team develops and baselines the software-related contractual requirements and places them under change control early in the project, but not later than release of the solicitation package. 3. The project team appraises system requirements change requests for their impact on the software being acquired. 4. The project team appraises all changes to the software-related contractual requirements for their impact on performance, architecture, supportability, system resource utilization, and contract schedule and cost. 5. Bi-directional traceability between the software-related contractual requirements and the contractor's software work products and services is maintained throughout the development.

SW-CMM Key Process Areas: Requirements Management and Software Product Engineering

KPA - purpose	***Goals***	***Activities***
Requirements Management (RM) - to establish a common understanding between the customer and the software projects of the customer's requirements that will be addressed by the software project. (This agreement is the basis for planning (see SPP) and managing (see SPTO) the project)	1. System requirements allocated to software are controlled to establish a baseline for software engineering and management use. 2. Software plans, products, and activities are kept consistent with the system requirements allocated to software.	1. The software engineering group reviews the allocated requirements before they are incorporated into the software project. 2. The software engineering group uses the allocated requirements as the basis for software plans, work products, and activities. 3. Changes to the allocated requirements are reviewed and incorporated into the software product.

KPA - purpose	***Goals***	***Activities***
Software Product Engineering (SPE) - to consistently perform a well-defined engineering process that integrates all the software engineering activities to produce correct, consistent software products effectively and efficiently. (SPE describes the technical activities of the project, e.g. requirements analysis, design, code, and test.)	1. The software engineering tasks are defined, integrated, and consistently performed to produce the software. 2. Software work products are kept consistent with each other.	*1. Appropriate software engineering methods and tools are integrated into the project's defined software process*. 2. The *software requirements* are developed, maintained, documented, and verified by systematically analyzing the allocated requirements according to the project's defined software process. 3. The software design is developed, maintained, documented, and verified, according to the project's defined software process, to accommodate *the software requirements* and to form the framework for coding. 4. The software code is developed, maintained, documented, and verified, according to the project's defined software process, to implement *the software requirements* and software design. 5. Software testing is performed according to the project's defined software process. 6. Integration testing of the software is planned and performed according to the project's defined software process. 7. System and acceptance testing of the software are planned and performed to demonstrate that *the software satisfies its requirements.* 8. The documentation that will be used to operate and maintain the software is developed and maintained according to the project's defined software process. 9. Data on defects identified in peer reviews and testing are collected and analyzed according to the project's defined software process. 10. Consistence is maintained across software work products, including the software plans, process descriptions, *allocated requirements, software requirements*, software design, code, test plans, and test procedures.

A2. Systems Engineering, System Procurement, and Planning and Management in the AMS Process Charts (draft of Sept. 12, 1996)

The following is extracted from the draft AMS Process Charts of September 12, 1996. The charted process is transformed into a work breakdown structure for ease of analysis and presentation here. Three "activity strings" have been extracted: "Planning and Management" , "System Procurement", and "Systems Engineering". These are from the process charts associated with the acquisition of systems and software, where the solution is to acquire NDI hardware and developmental software. (The charts do not pertain to Mission Analysis or Investment Analysis phases of AMS).

These charts assume the existence of an "Investment Analysis Report", a "Preliminary Acquisition Strategy Paper (ASP)," and a "Program Requirements Baseline (PRB)" when the processes commence. *(Note: The documents don't say who prepared the preliminary ASP or the PRB; perhaps the PRB is the Acquisition Program Baseline produced during Investment Analysis ??)*

1.0 Plan Program

1.1 Planning and Management

1.1.1 Plan acquisition strategy
- 1.1.1.1 Analyze IA Report/Verify LCCE *
- 1.1.1.2 Select system development model
- 1.1.1.3 Tailor acquisition process *

1.1.2 Prepare Acquisition Strategy Paper (ASP)

1.1.3 Plan Program Management Strategy
- 1.1.3.1 Formulate Rqmts Mgt Strategy (IPP)
- 1.1.3.2 Obtain support and other resources
- 1.1.3.3 Establish IPT
- 1.1.3.4 Formulate Risk Mgt strategy (IPP)
- 1.1.3.5 Develop Integrated Program Schedule (IPP)

1.1.4 Prepare IPP

1.5 System Procurement

1.5.1 Develop Procurement Approach (IPP)
- 1.5.1.1 Formulate contract tracking and oversight strategy
- 1.5.1.2 Select type of contract
- 1.5.1.3 Determine requirements for technical reviews
- 1.5.1 4 Prepare source selection plan
- 1.5.1.5 Prepare government cost estimate

1.5.2 Prepare Procurement Package

1.6 System Engineering

1.6.1 Analyze Performance Requirements and Constraints
- 1.6.1.1 Define hw life cycle
- 1.6.1.2 Define sw life cycle

1.6.2 Plan Engineering Mgmt Activities
- 1.6.2.1 Plan integration activities
- 1.6.2.2 Identify cost and schedule drivers; technical risk factors

  - 1.6.2.3 Plan Risk Reduction activities
  - 1.6.2.4 Plan hw management activities
  - 1.6.2.5 Plan sw management activities
  - 1.6.2.6 Plan engineering specialty management activities
- 1.6.3 Prepare inputs to procurement package
  - 1.6.3.1 Identify engineering documentation requirements
  - 1.6.3.2 Identify hw SOW, CDRLs, STDs, Eval Factors
  - 1.6.3.3 Identify sw SOW, CDRLs, Schedules, Eval Factors
  - 1.6.3.4 Identify specialty SOW, CDRLs, Schedules, Eval Factors
- 1.6.4 Draft Functional Spec
  - 1.6.4.1 Identify Interface Requirements

2.0 Obtain Solution

2.1. Planning and Management

- 2.1.1 Manage Development Program
  - 2.1.1.1 Manage development contract
    - 2.1.1.1.1 Joint SW management reviews
  - 2.1.1.2 Assess and Report Progress
  - 2.1.1.3 Manage Program Baseline
  - 2.1.1.4 Review and Update Resource Allocation Plans
  - 2.1.1.5 Acquire and Allocate Resources
- 2.1.2 Plan for deployment
  - 2.1.2.1 Plan Implementation activities
  - 2.1.2.2 Determine Readiness for Deployment (DRR)
- 2.1.3 In-service Decision
- 2.1.4 Manage Production Program
  - 2.1.4.1 Obtain and Allocate Resources
  - 2.1.4.2 Manage Production Contract

2.5 System Procurement

- 2.5.1 Award and administer development contract
  - 2.5.1.1 Release SIR
  - 2.5.1.2 Evaluate bids and negotiate
  - 2.5.1.3 award contract
    - 2.5.1.3.1 Provide GFI and GFE
  - 2.5.1.4 Baseline contractor software and engineering plans
- 2.5.2 Accept first and subsequent articles
- 2.5.3 Production decision
- 2.5.4 Award and administer production contract
  - 2.5.4.1 Exercise production option

2.6 System Engineering

- 2.6.1 Conduct Engineering Development Activities
  - 2.6.1.1 Perform risk reduction activities
  - 2.6.1.2 Administer Requirements baseline
    - 2.6.1.2.1 Control hw requirements changes
    - 2.6.1.2.2. Control sw requirements changes
    - 2.6.1.2.3 CM: CCB activities
  - 2.6.1.3 System requirements analysis (*return here for evolutionary devel)
    - 2.6.1.3.1 Identify hw requirements and obtain NDI hw

2.6.1.3.2 Adapt NDI hw
2.6.1.3.3 SW requirements analysis (SSR) (*return here for incr devel)
2.6.1.3.4 Preliminary/detailed design (PDR/CDR)
2.6.1.3.5 CM: allocated baseline
2.6.1.3.6 Engineering specialties: evaluate NDI products
2.6.1.4 System Design
2.6.1.5 System integration and testing
2.6.1.5.1 Coding & CSU testing/CSC integ & testing/CSCI testing(TRR)
(* return to system or software requirements analysis)
2.6.1.5.2 DT & E
2.6.1.6 Fabricate production representative units
2.6.1.7 Test and resolve deficiencies
2.6.1.7.1 Prepare site acceptance test plans and procedures
2.6.1.7.2 test: OT&E
2.6.1.8 FCA/PCA
2.6.1.8.1 FCA/PCA: HW product specs
2.6.1.8.2 FCA/PCA: SW product specs
2.6.1.8.3 CM: Product baseline
2.6.2 Conduct Production Engineering Activities
2.6.2.1 Plan hw and sw lifecycle activities
2.6.2.2 Administer Production changes
2.6.2.2.1 Incorporate hw changes
2.6.2.2.2 Incorporate sw changes
2.6.2.2.3 CM: Manage Product baselines
2.6.2.2.4 Test: PAT&E

3.0 Implement Solution

3.1 Planning and Management

3.1.1 Manage transition activities
3.1.1.1 Allocate, provide, and acquire resources
3.1.1.2 Transfer staff

3.5 System Procurement

3.5.1 Award Maintenance Contracts

3.6 System Engineering

3.6.1 Support Site acceptance testing
3.6.1.1 Support installation & checkout
3.6.1.2 Correct site specific deficiencies
3.6.2 Commission subsystem

4.0 Manage Solution During Operational Life Cycle

4.1 Planning and Management

4.1.1 Manage in-service subsystem
4.1.1.1 Plan and manage acquisitions of upgrades
4.1.1.2 Manage maintenance activities
4.1.1.3 Allocate, provide and acquire resources
4.1.1.3.1 Assess adequacy of sustainment funding
4.1.2 Plan for decisions to upgrade, replace, or remove

4.1.2.1 Analyze cost & performance data

4.1.2.2 Formulate recommendations for upgrade/replace decisions

4.1.2.2.1 Upgrade decision

- outside baseline, go to investment analysis
- in baseline: go to plan program

4.1.2.2.2 Replacement decision - go to investment analysis

4.1.2.2.3 Removal decision - go to remove solution

4.5 System Procurement

4.5.1 Maintain Contractor Support

4.5.2 Obtain upgrades (loop back to obtain solution)

4.6 Systems Engineering

4.6.1 Monitor and report subsystem performance

4.6.1.1 Identify operational performance shortfalls

4.6.1.1.1 Monitor hw performance

4.6.1.1.2 Monitor sw performance

4.6.1.1.3 (Eng Specialties) Monitor system performance

4.6.2 Maintain subsystem

4.6.2.1 Remove latent defects

4.6.2.2 Implement changes to correct systemic problems

4.6.2.3 Maintain hw (restore functionality)

4.6.2.4 Maintain SW (correct latent errors)

4.6.3 Implement planned upgrades or new requirements

4.6.3.1 Assess impact of upgrades

4.6.3.2 Design, integrate & test upgrades

4.6.3.3 Upgrade hw

4.6.3.4 Upgrade sw

4.6.3.3 CM: manage HW, SW configuration

4.6.3.4 Test: test upgrades

4.6.4 Seek technology opportunities to enhance capability or reduce cost

5.0 Remove Solution

5.1 Planning and Management

5.1.1 Execute removal plans

5.1.1.1 Allocate resources to removal activities

5.5 System Procurement

5.5.1 Award and Administer contracts

5.5.2 Dispose of removed equipment

5.6 System Engineering

5.6.1 Manage NAS Configuration